Hierarchical graph maps for visualization of collaborative recommender systems

Antonio Hernando

Ricardo Moya

Fernando Ortega

Jesús Bobadilla

1. Introduction

Recommender systems (RS) [1] are developed to attempt to reduce part of the information overload problem produced on the Net. As opposed to other traditional help systems, such as search engines (Google, Yahoo, etc.), RS base their operations on different types of filtering: collaborative, demographic, content-based, social, knowledge based, hybrid, etc. The collaborative filtering (CF) process has the main importance in modern RS; CF is based on making predictions about a user's preferences or tastes based on the preferences of a group of users that are considered similar to this user.

RS cover a wide variety of applications [2–8]. Memory-based CF methods [9–11] are based on a rating matrix where the votes from users on items are recorded into a database; this database is defined using a row for each RS user and a column for each RS item. RS matrices are sparse, since users only vote on a limited number of the available items. Memory-based CF methods use similarity metrics [12–14] that act directly on the rating matrix; these metrics mathematically express a distance between two users based on each of their ratios. Matrix sparsity is a serious drawback using CF similarity metrics [15]. CF can be improved in several areas, such as to reflect fluctuations in users' behaviour over time [16].

Similarity metrics and measures can be used to calculate similarities between users – user-to-user CF – or between items – item-to-item CF [17]. Most of the papers published in the area of CF focus on user-to-user CF, because these algorithms provide more accurate results than those based on item-to-item.

An enormous number of papers have been published in the CF RS area, covering a range of specific issues. Most of the papers propose metrics, methods and algorithms for improving predictions and recommendation accuracy, coverage, precision/recall, etc.; nevertheless, there are other purposes that are taken into account, such as avoiding overspecialization phenomena, finding good items, trusting recommendations, novelty, precision and recall measures, sparsity, cold start issues, incorporating social information and framework definition.

The representation and visualization of RS is an area which has not been covered in academic publications. To date, only one paper has been published on a specific way of using the memory-based information (users' votes on the items) from a visual information retrieval approach: Hernando et al. [18] present a novel technique, based on visualization of trees of items, for explaining recommendations. Moreover, a remarkable unpublished result has been discovered within the environment of the company Netflix (Ensemble) [19]. This group provides visualization results with a very similar appearance to those set out in our paper, but which display less visual information than the ones we provide.

Our approach contains the following improvements compared with the Ensemble results: (a) we provide all the formalisms and equations to obtain the paper's results; (b) the similarity metric used in this paper obtains a suitable and limited number of items related to any node (out of the nodes graph); (3) edge distances and edge colours are defined according to well-known metrics combined into a reputed CF similarity metric – JMSD; and (4) nodes importance are represented proportionally to their popularity.

Owing to the lack of publications that provide visualization results of the memory-based information contained in RS, the academic starting point must be based on a more classical and general focus: graph visualization [20–22]. When the size of the graphs is very large, as is the case in RS, the approach used for their visualization is to convert the graph into a hierarchical graph map [23, 24]. To carry out this conversion, any of the well-known minimum spanning forest algorithms from the weighted graph can be used.

At present, there are a large number of publications in which hierarchical graphs are provided based on a completely different data source from the one used by RS – phylogenetic data. Phylogenomics is aimed at studying functional and evolutionary aspects of genome biology using phylogenetic analysis of whole genomes [25, 26]. Although phylogenetic trees share important characteristics with RS-based trees, we must take into consideration that the first ones include an inheritance relationship between the nodes of the hierarchical structure, while in RS-based trees this does not occur.

The rest of the paper is structured as follows: Section 2 describes the proposed visualization method; Section 3 reports the evaluation results; and Section 4 summarizes this work.

2. Method

2.1. Introduction

The starting information is a matrix R of U users and I items, in which the users' votes on the items appear. As thousands of items commonly exist, each user only votes for a small proportion of them, and therefore the matrix presents a high degree of sparsity. The database with which we carry out the experiments is MovieLens 1M, which contains one million votes (ranging from 1 to 5) cast by 6040 users on 3900 movies.

The objective of this paper is to provide a method that visualizes the similarity relationships that exist between the items of the CF RS, as well as the relative importance of each of these. The starting data used is the values of the votes (and lack of votes) contained in the matrix $R_{U,I}$. Using the same method, the relationships between the users can be visualized by applying it to the rows of users instead of to the columns of items contained in the matrix R.

The importance of each item can be easily determined from the number of votes the item has obtained and the rating of these votes, in such a way that a highly voted and positively rated movie will be considered important. The equation we have designed to determine the importance of each item $i \in I$ is:

$$\text{importance}(i) = \sum_{u \in U} [r_{u,i} - [min + (max - min)/2]] \tag{1}$$

where min in the minimum allowed vote (1 using Movielens) and max is the maximum allowed vote (5 using Movielens).

The relationships that exist between the items are determined by applying one of the published similarity measures and metrics [2, 7, 8, 14]. The metric that we have selected is JMSD [6], owing to its good accuracy results and the possibility of breaking down its operation into structural similarity information and numerical similarity information.

JMSD is a metric, and therefore $JMSD_{item x,item y} = JMSD_{item y,item x}$ is true, that is, the similarity between the item x and the item y is the same as the similarity between the item y and the item x. By applying the JMSD similarity metric between each pair of items of the database, we obtain the similarities matrix $S_{I,I'}$. The upper triangular matrix of S is symmetrical to the lower triangular matrix and therefore it will only be necessary to process and store half of its elements $(S_{I,I'} \mid I < I')$.

Matrix S forms a graph that could be visually represented if the number of items were small. In the RS this circumstance is not produced because the number of items (movies, music, videogames, books, etc.) is always in the thousands. As we explained in Section 1, the most suitable approach is to convert the graph of similarities into a hierarchical graph; to do this, we have used the classic minimum spanning forest algorithm of Prim, obtaining the hierarchical structure that we will call RS-IST (Recommender System Items Similarities Tree). Similarly, by processing users instead of items, we would obtain an RS-UST (Recommender System Users Similarities Tree).

2.2. Formalism

RS-IST can be formulated as a multi-graph, represented by a triplet $G = (V, E, m)$ where V is the vertex set, E a subset of $V \times V$ is the set of edges and $m: E \to R^+$ is a function that assigns to each edge a non-negative multiplicity.

In a first approach, the RS-IST obtained could be visualized in a simpler way: representing the tree structure without taking into consideration the similarity values between the items; that is, without using the values of function m of the multi-graph $G = (V, E, m)$. In this paper we propose a visual representation of the RS-IST which, in addition to the tree structure, provides:

- the importance of each of the items $(\forall v \in V, p : V \to R^+)$ the numerical similarity between each pair of connected items $(\forall e \in E, n : E \to R^+)$ structural similarity between each pair of connected items $(\forall e \in E, s : E \to R^+)$

In this way, our RS-IST is represented by the multi-graph $G = (V, E, p, n, s)$.

- E is determined using the similarity metric JMSD [6]:

$$JMSD_{x,y} = Jaccard_{x,y}\left(1 - MSD_{x,y}\right) \tag{2}$$

where

$$Jaccard_{x,y} = \frac{\left|U^*\right|}{\left|\left\{u \in U | r_{u,x} \neq \varnothing \vee r_{u,y} \neq \varnothing\right\}\right|} \tag{3}$$

$$MSD_{x,y} = \frac{1}{\left|U^*\right|} \sum_{u \in U^*} \left(r_{u,x} - r_{u,y}\right)^2, \tag{4}$$

and

$$U^* = \left\{u \in U | r_{u,x} \neq \varnothing \wedge r_{u,y} = \varnothing\right\} \tag{5}$$

- p is determined using *importance*(i), where i is a vertex of G (1);
- n is determined using $MSD_{x,y}$ (4);
- s is determined using $Jaccard_{x,y}$ (3).

While $MSD_{x,y}$ provides us with the classic numeric similarity between two items (as obtained using the metrics Pearson correlation or adjusted cosine), $Jaccard_{x,y}$ offers us a measure of reliability of the similarity numerical value. By way of example, two items x (with 900 ratings) and y (with four ratings) can have a numeric similarity value of 0.9 on a scale of 0–1. In the same example, x and z (with 800 ratings) can also have a numeric similarity value of 0.9. While in the first case the value of Jaccard will be very low, in the second case, it will probably be much higher, which indicates what we intuitively perceived: the numeric similarity value of x and z is much more representative than that of x and y.

2.3. Graphical representation

The representation of multi-graph $G = (V, E)$ allows us to draw all the edges belonging to set E with the same length and all the vertices belonging to set V with the same size; furthermore, it is not necessary to use colours to represent characteristics of the tree. The representation of multi-graph $G = (V, E, m)$ requires each edge belonging to E to be drawn with the value obtained by applying the function $m: E \rightarrow R^+$, which would indicate the similarity between items. This is the approach offered by Ensemble. Although the values of m can be represented with the lengths of the edges, Ensemble makes use of colours: 'Similar movies are connected by a line. Line colours closer to red indicate a weaker similarity, and colours closer to yellow indicate a stronger similarity'.

In this paper, we propose the representation of items of an RS using multi-graph $G = (V, E, p, n, s)$:

- The values provided by function p are visualized using different sizes (circular) on the tree's vertex.
- The values provided by function n (numerical similarities) are visualized using different lengths of the edges. The more numerically similar the two items x and y are, the closer they are represented; that is, the smaller the value of $MSD_{x,y}$ is, the closer x and y are visualized.
- The values provided by function s (structural similarities) are visualized using different colours on the edges. The more structurally similar the two items x and y are, the more representative the colour used on the scale is; that is, the larger the value of $\text{Jaccard}_{x,y}$, the greater the colour chosen on the scale to represent the edge which joins items x and y.

The functions p, n and s require a stage which brings the distributions of equations (1), (3) and (4) closer in uniform or normal distributions. This is necessary to ensure that there are no extreme variations in the lengths and the colours of the edges, as well as in the sizes of the vertex. Starting from the information of the standard deviation and the average of p, n and s, we have created the functions of normalization: $f(p)$, $g(n)$ and $h(s)$. Figure 1 shows the original and transformed functions p, n and s.

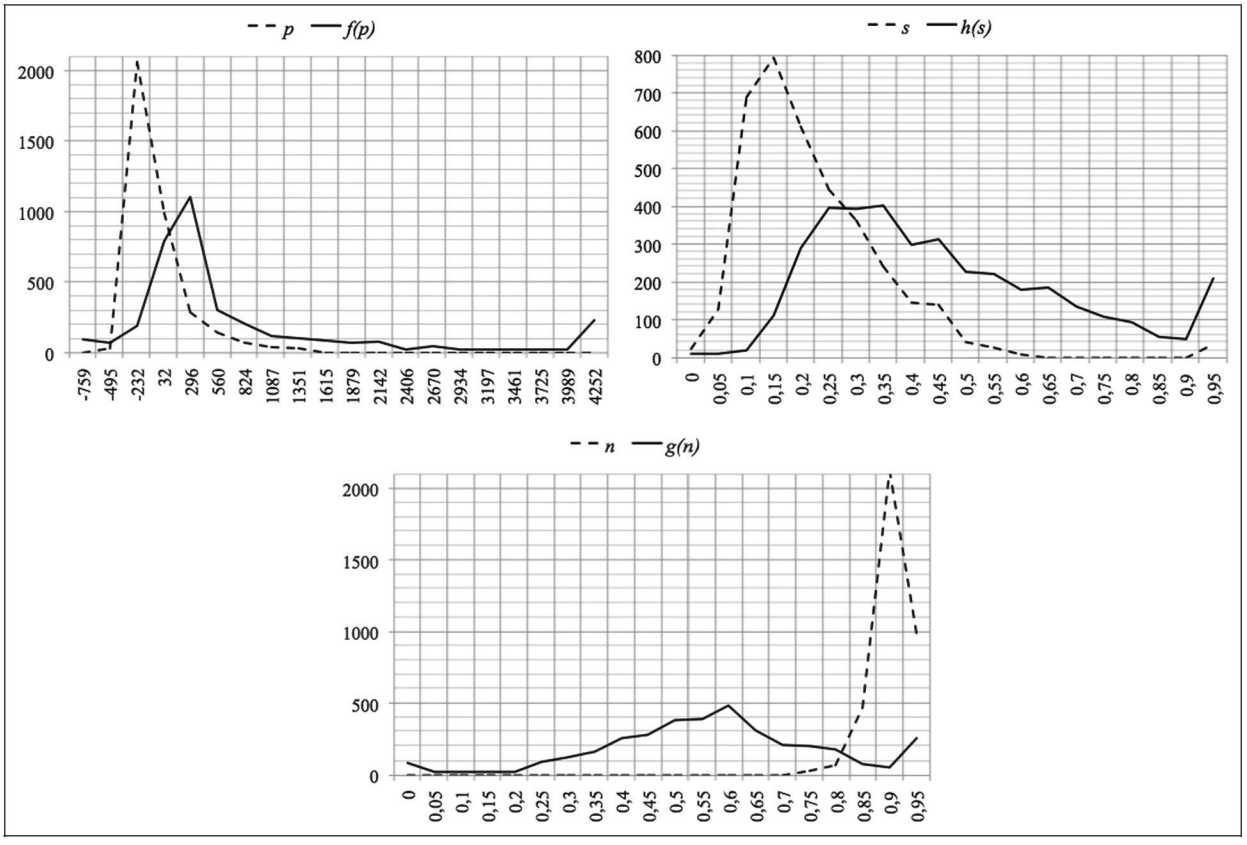

Figure 1. Original and transformed functions p, n and s.

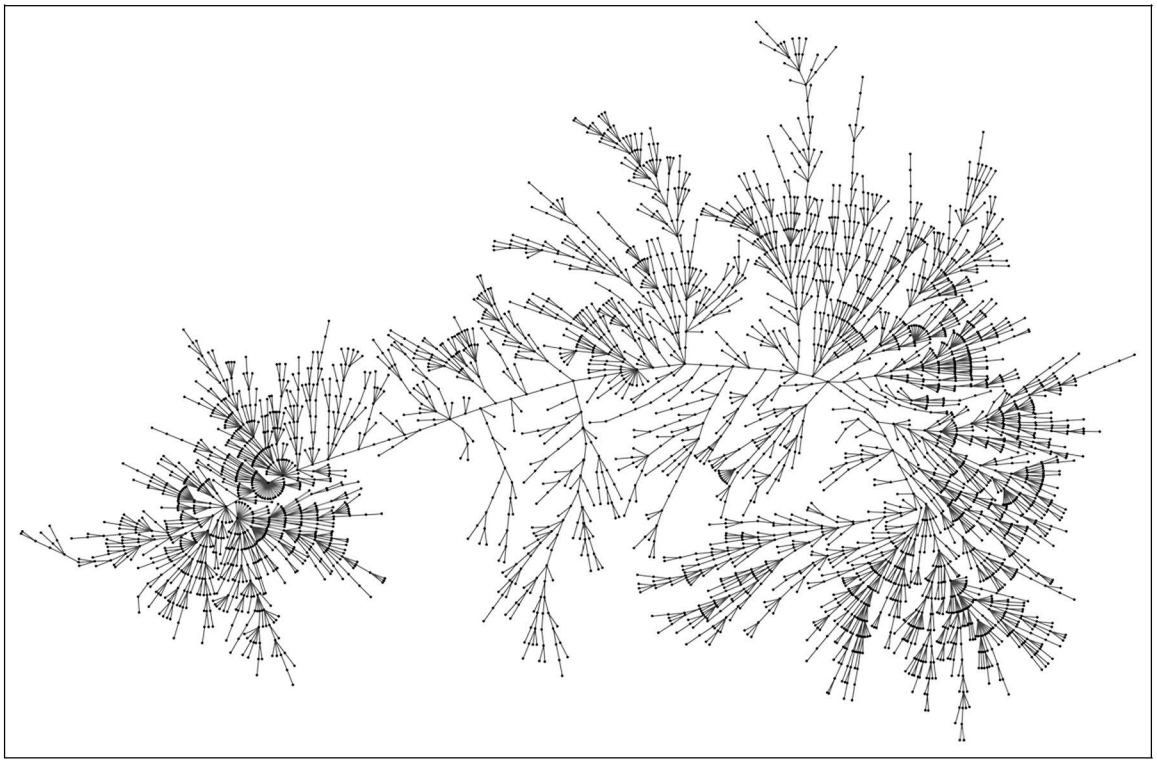

Figure 2. Multi-graph $G = (V, E)$ representation.

3. Results

A first approach for the visualization of the items of an RS consists in representing the multi-graph $G = (V, E)$. The graph obtained (Figure 2) shows the structure of the minimum spanning tree; on this tree we can consult which set of items has been rated the most similar to a given one: we just need to observe their adjacent items and continue, if required, with the rest of the branches that start from their adjacent items.

The representation proposed in this paper uses the information contained in the multi-graph $G = (V, E, p, n, s)$. We represent: (a) the importance of each of the items (using different sizes of circles on the tree's vertex); (b) numerical similarity between each of the connected items (using different lengths of the edges); and (c) structural similarity between each pair of items connected (using different colours on the edges). Figure 3 shows the result. Figure 3 represents the vertex using black circles. The most representative structural similarity (greater similarity) corresponds to the light blue colour, followed by blue, dark blue, purple, red and orange.

Figure 4 shows a section of the most representative area of RS-IST (right zone of Figure 3). In this zone there are a large number of important items (larger black circles). These important items tend to be related to each other with short edges (high numerical similarity) and light blue, blue and dark blue colours (high structural similarity), forming the 'backbone' of the RS-IST. In addition, the OUT of the important vertex remains small, making it easier to identify its directly related items.

From the 'backbone' of the RS-IST, the branches formed by less important items extend, which are less important as they get closer to the leaf items, as occurs in nature. This trend is reinforced by the progressive decrease in the numerical and structural similarities and the progressive increase in the OUT as the items move closer to the ends (leaves). Figure 4 gives a better view of the reduction in size of the vertex, while Figure 3 gives a better view of the evolution in the scale of colours and in the lengths of the edges.

Figure 5 shows a section of the zone with less important items (left zone of Figure 3). The size of the vertex is the minimum that can be represented, the predominance of warm colours indicates very low structural similarity and the length of the edges tends to be greater than in Figure 4 (zone of most important items). The vertex with high OUTs, represented as a light blue circle, corresponds to items that have votes from very few users; as it has very few votes, there is a greater probability of finding other items that display a high coincidence in the votes of this small set of users.

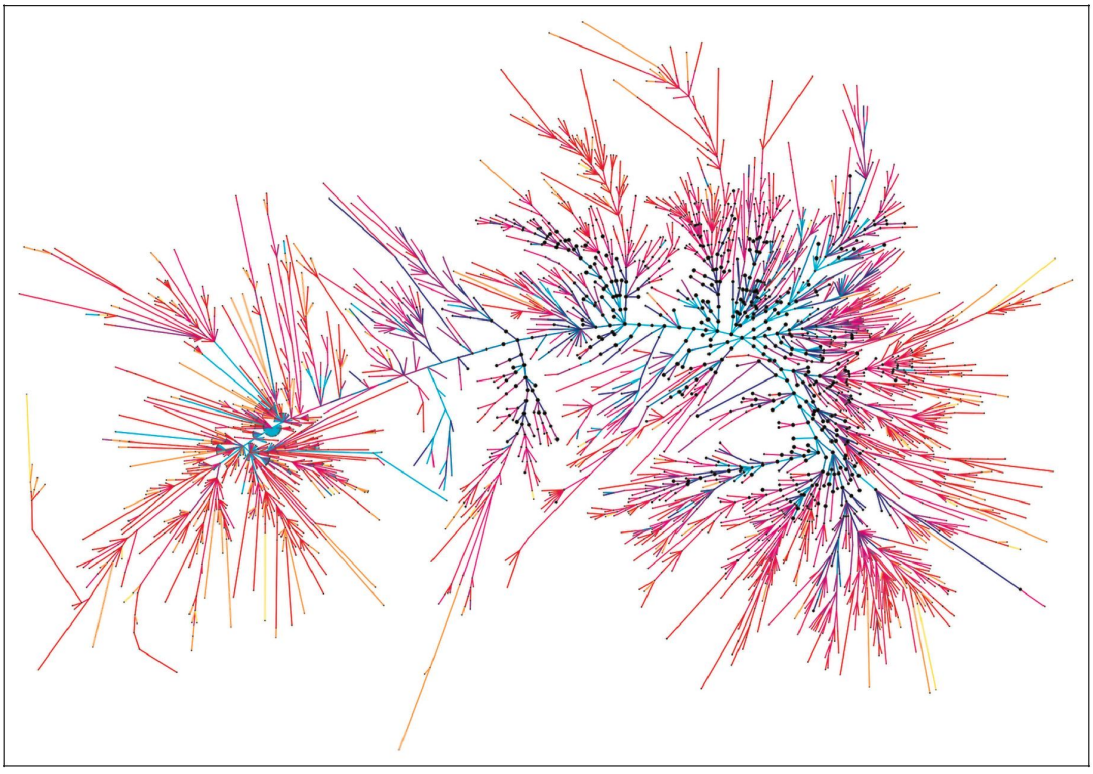

Figure 3. Multi-graph $G = (V, E, p, n, s)$ representation. The article is published in colour online at http://jis.sagepub.com/

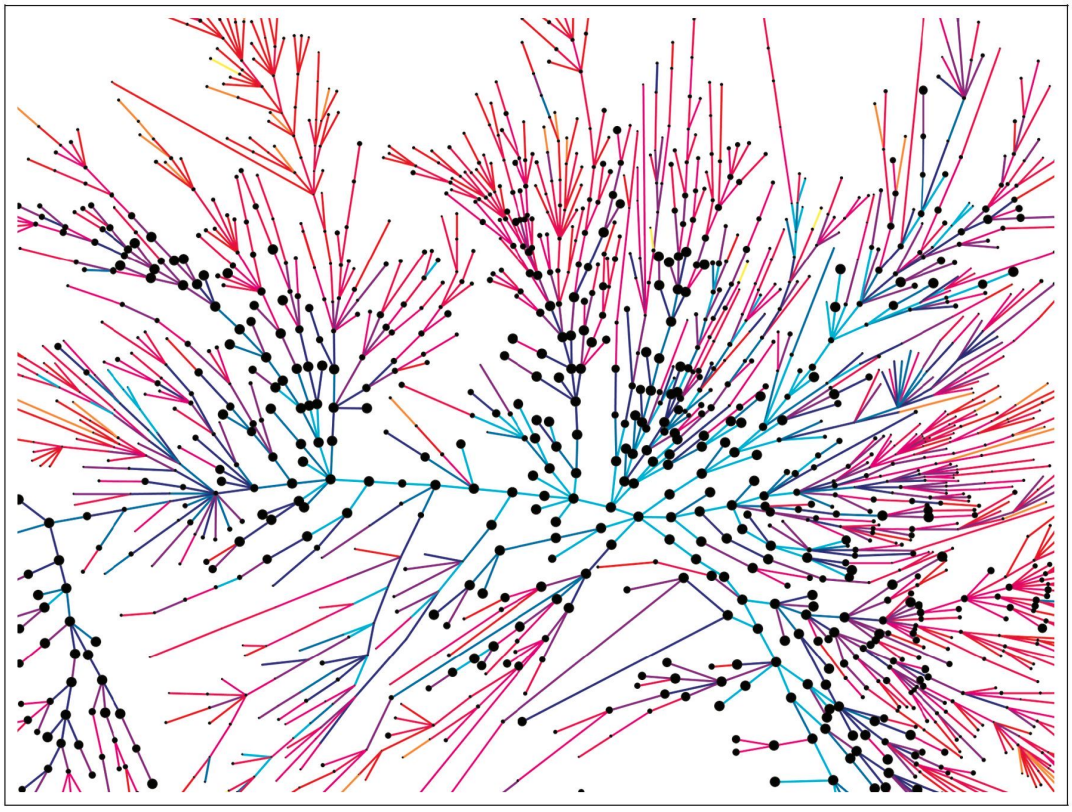

Figure 4. Zone that contains the most important items. The article is published in colour online at http://jis.sagepub.com/

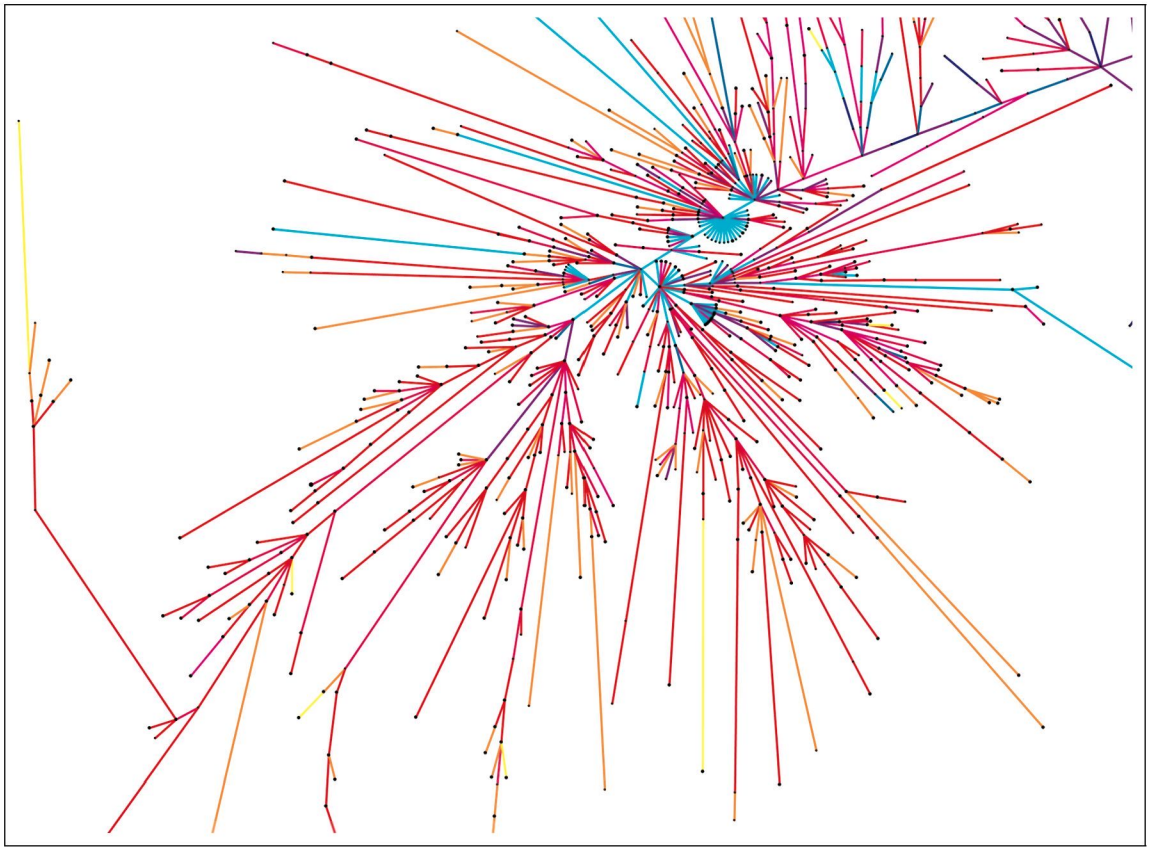

Figure 5. Zone that contains the least important items. The article is published in colour online at http://jis.sagepub.com/

The improved visualization of the items represented in Figure 5 can help to open an interesting field in the area of CF RS: the use of metrics and similarity measures adapted to these cases, such as the new user cold-start metrics. In this way, the RS-IST proposed in the paper will visually demonstrate the integrity of each item to item cold-start metric.

An important characteristic of the RS-IST proposed is their visual scalability: as the RS is used more, the number of votes that the items receive from the users tends to grow. This way, as the RS is used more, the structure and scale of colours of the RS-IST will tend to distribute themselves in the shape of a tree in which it is easier to distinguish the differences between the most important branches and the least important branches (terminal branches). In the same way, it is easier to distinguish the differences between the most important vertices (tree trunk) and the least important vertices (vertex leaves).

Figure 6 shows the details of a small portion of Figure 3. By analysing the connections between the movies, we can see how the RS-IST is capable of adequately grouping and relating the items. We can observe how the proposed method and metrics establish a clear neighbourhood relationship between movies of the same genre; what is more, even movies that belong to the same saga (Terminator, Star Wars, Indiana Jones, etc.) are considered to be directly related.

RS-IST can be generated by selecting any item as a root vertex. It is also possible to restrict the representation to a maximum depth, which facilitates their visualization and subsequent analysis by RS users. Figure 7 shows the RS-IST obtained by taking as a root vertex 'Star Wars: Episode V' and depth 2 (top left-hand corner of Figure 7) or depth 6.

4. Conclusions

Collaborative filtering recommender systems can be visually represented as hierarchical graph maps; the representation accepts the visualization of a tree of items or a tree of users. Regardless of whether the elements represented are items or users, the method proposed provides the structure of the tree, the sizes of the vertices, and the length and colour of the edges. The sizes of the vertices indicates their importance; the length of the edges indicates the similarity that exists between pairs of vertices; and the colour of the edges indicates the reliability of the value of similarity.

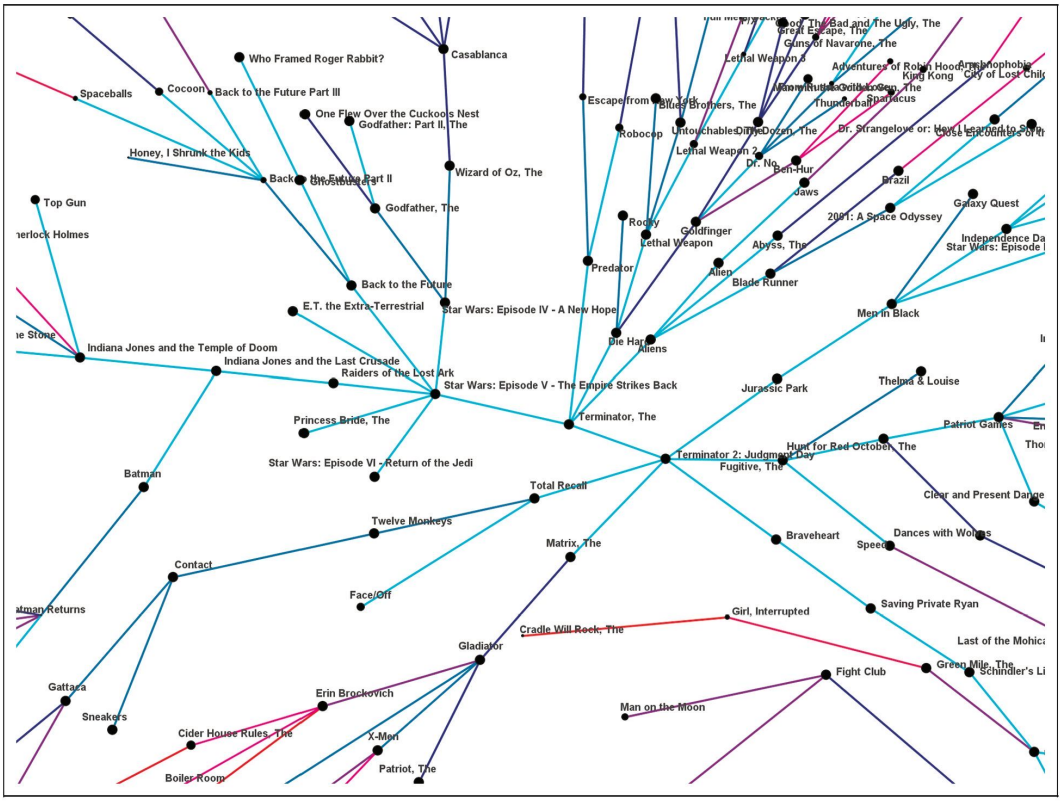

Figure 6. Detail of the movies tree. The article is published in colour online at http://jis.sagepub.com/

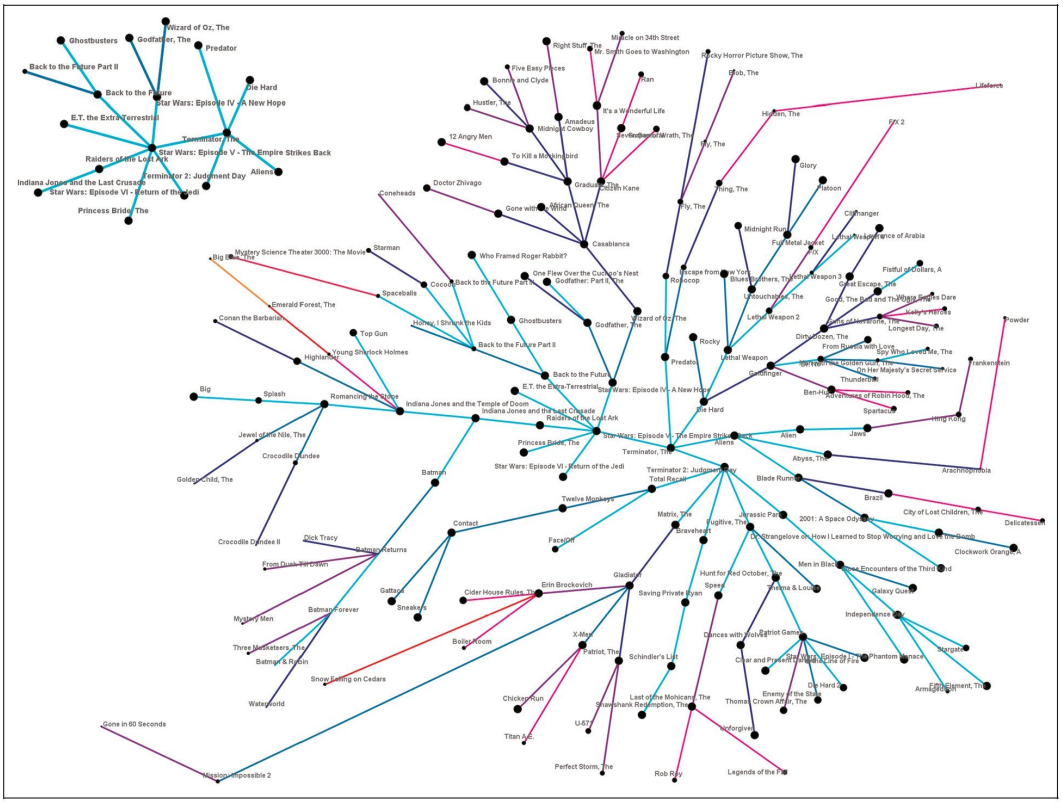

Figure 7. Tree generated by taking as a root vertex 'Star Wars: Episode V' and depth 2 (top left-hand corner of Figure 7) or depth 6. The article is published in colour online at http://jis.sagepub.com/

Using the database Movielens 1M, the visualization of the items obtained allows us to clearly identify the areas where the set of most significant movies is represented. Using the proposed method, the most relevant information is arranged in a limited set of central branches from which the tree ramifies until it reaches the tree leaves.

When we advance into the tree to see the environment of a particular item (in our case study a movie), we can easily determine which items have been most similarly rated to it: the closest ones and the ones connected with the most representative colours. Using Movielens 1M, each movie presents a suitable and sufficient number of close movies that are related to it as regards theme, genre, sequel, director, etc.

This paper presents a wide range of possibilities to be developed as future works: (a) study of the characteristics of the users tree, personalizing the results to each user that requires it; (b) introduction of various measures of reliability; (c) comparatison of the resulting tree structures on using various similarity metrics; (d) incorporation of a prior clustering stage; and (e) information retrieval in the different areas in which memory-based recommender systems exist (social networks, blogs, music, geographic recommender systems, etc.).

Acknowledgements

The authors acknowledge the Grouplens Research Group and to the FilmAffinity.com company. The authors also thankfully acknowledge the computer resources, technical expertise and assistance provided by the Supercomputation and Visualization Center of Madrid and the Spanish Supercomputation Network.

Funding

This research was supported by the Spanish Ministerio de Educación, Cutura y Deporte TIN2012-32682 project.

References

[1] Bobadilla J, Ortega F, Hernando A and Gutiérrez A. Recommender systems survey. *Knowledge Based Systems* 2013; 46: 109–132.

[2] Antonopoulus N and Salter J. Cinema screen recommender agent: Combining collaborative and content-based filtering. *IEEE Intelligent Systems* 2006: 35–41.

[3] Bobadilla J, Serradilla F and Hernando A. Collaborative filtering adapted to recommender systems of e-learning. *Knowledge Based Systems* 2009; 22: 261–265.

[4] Huang Z, Chung W and Chen H. A graph model for e-commerce recommender systems. *Journal of the American Society for Information Science and Technology* 2004; 55(3): 259–274.

[5] Liu DR, Lai ChH and Chen YT. Document recommendations based on knowledge flows: A hybrid of personalized and group-based approaches. *Journal of the American Society for Information Science and Technology* 2012; 63(10): 2100–2117.

[6] Porcel C and Herrera-Viedma E. Dealing with incomplete information in a fuzzy linguistic recommender system to disseminate information in university digital libraries.*Knowledge-Based Systems* 2010; 23(1): 32–39.

[7] Serrano J, Viedma EH and Olivas JA. A google wave-based fuzzy recommender system to disseminate information in University Digital Libraries 2.0. *Information Sciences* 2011; 181(8): 1503–1516.

[8] Zheng N, Qiudan L, Liao S and Zhang L. A comparative study of recommendation algorithms in Flickr. *Journal of Information Science* 2010; 36(6): 733–750.

[9] Adomavicius G and Tuzhilin A. Toward the next generation of recommender systems: A survey of the state-of-the-art and possible extensions. *IEEE Transactions on Knowledge and Data Engineering* 2005; 17(6): 734–749.

[10] Breese JS, Heckerman D and Kadie C. Empirical analysis of predictive algorithms for collaborative filtering. In: *14th Conference on uncertainty in artificial intelligence*, 1998, pp. 43–52.

[11] Herlocker JL, Konstan JA, Riedl JT and Terveen LG. Evaluating collaborative filtering recommender Systems. *ACM Transactions on Information Systems* 2004; 22(1): 5–53.

[12] Bobadilla J, Serradilla F and Bernal J. A new collaborative filtering metric that improves the behavior of recommender Systems. *Knowledge-Based Systems* 2010; 23(6): 520–528.

[13] Bobadilla J, Hernando A, Ortega F and Gutiérrez A. Collaborative filtering based on significances. *Information Sciences* 2011; 185(1): 1–17.

[14] Bobadilla J, Ortega F and Hernando A. A collaborative filtering similarity measure based on singularities. *Information Processing and Management* 2012; 48(2): 204–2017.

[15] Hoseini E, Hashemi S and Hamzeh A. SPCF: a stepwise partitioning for collaborative filtering to alleviate sparsity problems. *Journal of Information Science* 2012; 38(6): 578–592.

[16] Rafeh R and Bahrehmand A. An adaptive approach to dealing with unstable behaviour of users in collaborative filtering systems. *Journal of Information Science* 2012; 38(3): 205–221.

[17] Sarwar B, Karypis G, Konstan J and Riedl J. Item-based collaborative filtering recommendation algorithms. In: *World Wide Web conference* 2001, pp. 285–295.

[18] Hernando A, Bobadilla J, Ortega F and Gutiérrez A. Trees for explaining recommendations made through collaborative filtering. *Information Sciences* 2013; 239(1): 1–17.
[19] Ensemble, http://www.the-ensemble.com/content/netflix-prize-movie-similarity-visualization
[20] Herman G and Melancon MS. Graph visualization and navigation in information visualization: A survey. *IEEE Transactions on Visualization and Computer Graphics* 2000; 6(1): 24–43.
[21] Von Landesberger T, Kuijper A and Schreck T. Visual analysis of large graphs: State of the art and future research challenges. *Computer Graphics Forum* 2011; 30(6): 1719–1749.
[22] Michailidis G. Data visualization through their graph representations. In: *Handbook of data visualization*. Berlin: Springer, 2008, pp. 103–120.
[23] Abello J. Hierarchical graph maps. *Computers and Graphics* 2004; 28: 345–359.
[24] Julien ChA, Tirilly P, Leide JE and Gustavino C. Constructing a LCSH tree of a science and engineering collection. *Journal of the American Society for Information Science and Technology* 2012; 63(12): 2405–2418.
[25] Dagan T. Phylogenomic networks. *Trends in Microbiology* 2011; 19(10): 483–491.
[26] Hsieh SY and Huang ChW. An efficient strategy for generating all descendant subtree patterns from phylogenetic trees with its implementation. *Applied Mathematics and Computation* 2007; 193: 408–418.